



# Semantic Representation For Navigation In Large-Scale Environments

Romain Drouilly, Patrick Rives, Benoit Morisset

## ► To cite this version:

Romain Drouilly, Patrick Rives, Benoit Morisset. Semantic Representation For Navigation In Large-Scale Environments. IEEE Int. Conf. on Robotics and Automation, ICRA'15, May 2015, Seattle, United States. hal-01122196

**HAL Id: hal-01122196**

**<https://inria.hal.science/hal-01122196>**

Submitted on 3 Mar 2015

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Semantic Representation For Navigation In Large-Scale Environments

Romain Drouilly<sup>1,2</sup>, Patrick Rives<sup>1</sup>, Benoit Morisset<sup>2</sup>

**Abstract**—Mimicking human navigation is a challenging goal for autonomous robots. This requires to explicitly take into account not only geometric representation but also high-level interpretation of the environment. In this paper, we demonstrate the capability to infer a route in a global map by using semantics. Our approach relies on an object-based representation of the world automatically built by robots from spherical images. In addition, we propose a new approach to specify paths in terms of high-level robot actions. This path description provides robots with the ability to interact with humans in an intuitive way. We perform experiments on simulated and real-world data, demonstrating the ability of our approach to deal with complex large-scale outdoor environments whilst dealing with labelling errors.

## I. INTRODUCTION

Autonomous navigation in real world environment is one of the key challenges to solve for mobile robots. They have to be able to find the best way from their current position to target position, whilst adhering to constraints and interacting with humans. To date, most of the strategies for route planning rely on metric or topological maps and constraints are expressed in terms of geometry, assuming the shortest path to be the best. While it is obviously useful to minimize the path length, depending on the scenario many other constraints have to be considered. Robots are required to deal with challenges that impose to redefine what "best" means. To name a few, contrarily to static scenes that represent a very small class of real-world environments, dynamic scenes contain moving objects with unpredictable behaviour and are subject to appearance changes. In such cases, an intelligent robot needs to consider those aspects when choosing the best route from one point to another. Other constraints like minimizing the collision risk or maximizing the stability of landmarks along the path are also crucial.

A recent trend in robotic mapping is to augment low-level maps with semantic interpretation of their content, which allows to improve the robot's environmental awareness. In mobile robot navigation, the so-called semantic maps have already been used to improve path planning methods, mainly by providing the robot with the ability to deal with human-understandable targets [1], [2]. However, despite their interesting results, these approaches rely on the "place" concept that is well-defined indoors but difficult to define outdoors.

In this work we propose to use high-level semantic interpretation of the scene both for specifying interesting targets and for choosing optimal path towards a given goal. To do so,

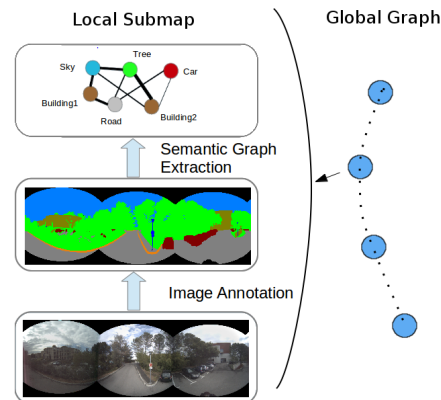


Fig. 1. Building Process of local submaps. Ego-centered spherical images are automatically labelled with Random Forest and CRF. Then a Semantic Graph is extracted from the labelled image.

we introduce a semantic cost function that takes into account high-level constraints through image annotation. This allows us to select a path according to criteria that capture scene features different from metrical data or topology but that are essential for robot navigation. Additionally, we propose a new semantic path representation that aims to address the problem of semantic path description in outdoor environments. Both this representation and the route planning algorithm rely on the hybrid map structure introduced in our previous work [3]. It uses only object position in images without the need of 3D scene reconstruction or the use of the ill-defined concept of "place" to describe robot positions.

The main contributions of this paper are twofold: a new framework to select the best route for the robot taking into account semantic content of the scene; and a new approach to describe a path in terms of semantics that easily adapt to both indoor and outdoor environments.

After a concise review of related work in section II, we present the Semantic Map structure in section III while our Semantic Route Planning framework and our Path Description methodology are introduced in section IV and V respectively. Then two series of experiments are presented on both simulated and real-world data in section VI, demonstrating the effectiveness of our algorithm.

## II. RELATED WORK

Semantics have been shown to be useful in inferring new information about the scene and to improve planning efficiency [4]. It is useful when the search space is large, translating the problem in a new space of smaller dimension.

<sup>1</sup>Authors are with INRIA Sophia-Antipolis, France  
romain.drouilly@inria.fr, patrick.rives@inria.fr

<sup>2</sup>Authors are with ECA Robotics, bmo@eca.fr

In the specific domain of mobile robot navigation, semantics has been used mainly for place modelling, allowing to specify high-level goals for robot navigation and description of trajectories. In [1] semantics is used to represent the environment as a graph of places. The robot reasons about space with this graph-based representation of places where semantics is embedded in the nodes while edges correspond to accessibility. A navigation framework based on high-level understanding of the world is proposed in [2]. As in the previous work, semantics are used both to define places and robot's actions related to navigation. Spherical images are labelled with Random Forest and a place recognition module is used that is based on SVM. The navigation behaviour is then described as a sequence of low-level behaviours organised in a hierarchical way. Similarly in [5] semantic regions and positions are defined to allow the robot to plan its path at a high-level. Each node of the graph is connected to others through edges encoding reachability of a place from another. Planning is done using metric constraints like the euclidean distance of two nodes.

The related problem of interpreting route from description has also received significant attention. For example a framework combining automatic semantic place labelling and human route description interpretation is presented in [6]. To handle the possible ambiguity that occurs due to perception errors, a probability distribution over the possible paths is built, and the best path is chosen computing the MAP of the distribution. In [7], a machine translation technique from natural language input is used to infer a most likely path within a Voronoi graph where the nodes of the graph correspond to different places like room, hall or doorways.

All these works offer high-level description for path planning. They model space as a set of semantic places connected topologically through robot's actions or metric constraints. However, all these works rely on the attribution of a label to each place to localize the robot. This is not feasible in outdoor environments where there is no clear separation between different places which is an issue that we specifically address in the proposed framework.

### III. SEMANTIC MAP FOR NAVIGATION

In this paper we address two problems: finding the best route according to high-level criteria and describing it using surrounding objects all along the trajectory. To address these problems, we rely on the map structure illustrated in fig 1 and a localization scheme described in our previous work [3]. For the sake of completeness we briefly review them in the following paragraph.

#### A. Hybrid Metric-Topologic-Semantic Map

Our map consists of a set of local submaps connected to each other in a graph whose edges are pairwise pose estimates. Those submaps are built from data captured by our multicamera stereovision system, described in [8]. It allows to capture ego-centered spherical views of the scene augmented with depth data. From the sequence of images acquired, we keep only reference keyframes as detailed in

[9]. That comprises the RGB-D level of submaps. Atop of this layer lies the *label layer* which is the result of the automatic annotation of the spherical views [3]. The scene parsing process occurs in two steps: first *Random Forests* are used to estimate the probability distribution over classes for each pixel. Then a *Fully Connected Conditional Random Field* corrects results taking into account neighbourhoods. Given a labelled image, a set of contiguous pixels with the same label is called a semantic area, denoted as  $A_i$ . It can contain several instances of each class depending of the perspective viewpoint. The contiguous semantic areas are connected to each other through edges  $E_{ij}$  encoding their relative position in image, constituting a *semantic graph*  $\mathbf{G} = \{A, E\}$ . This representation, is powerful as it encodes in a very compact way both the structure and the semantic content of the scene.

#### B. Semantic Content-based Localization

Following a route needs to solve the robot localization issue. An efficient algorithm for localization in our map has been developed in [3], where localization is formulated as a content request problem and allows to localize an image or a set of objects. The semantic graph is first extracted from the current annotated image. Then, the localization consists in comparing it with those of the submaps to estimate similarities. The comparison is done with an *interpretation tree* [10] matching one by one the nodes of the current semantic graph to those of the reference semantic graph using nodes properties and context. The similarity score  $\sigma$  between two semantic graphs  $G_1$  and  $G_2$  is measured as follows:

$$\sigma(G_1, G_2) = \exp^{1 - \frac{N}{N_m}} \quad (1)$$

where  $N_m$  is the number of nodes matched between the two graphs denoted as  $A_{12} = A_{G1} \cap A_{G2}$  and  $N$  the total number of nodes in the current semantic graph. The submap with the highest score  $\sigma$  corresponds to the most probable location.

### IV. SEMANTIC-BASED ROUTE PLANNING

A route is defined by a set of adjacent local submaps in the global graph. Once submaps corresponding to the current and the target positions of the robot are found, the route planning problem consists in finding the best path in the graph to go from one to the other. We want to optimize this path with respect to semantic criteria, therefore we need to use semantics to weight edges between nodes. To do this, we define a heuristic cost function that computes the weight of an edge depending on the semantic scene content. Two adjacent submaps share objects that are visible from the two viewpoints. The cost function relies on these objects. The highest is the weight, the more penalized will be the path. Several scene features are used to build the cost function, listed as follows:

a) *Number of shared objects*: To navigate within our map, the robot needs landmarks all along the way. If the environment changes considerably between two submaps, the risk for the robot to be lost is higher as it may not be able to localize itself. Therefore the higher is the number of objects

shared between two submaps the more it should be preferred. However this number strongly depends on the number of objects visible in a given scene. Therefore we compute the number of objects shared relatively to the total number of objects visible in the current scene. The *Shared Objects Score*  $S$  is:

$$S(G_1, G_2) = 1 - \sigma(G_1, G_2) \quad (2)$$

*b) Object class:* objects can be more or less helpful landmarks for navigation depending on their class. Cars are dynamic objects and therefore not well suited to characterize a place while buildings are stable landmarks. We give a weight to all classes according to their expected usefulness for purpose of navigation. The lowest is the penalizing weight, the more useful is the object. For an edge between two adjacent submaps, we compute the *Class Score*  $C$  as the average of the class weight of each shared object:

$$C(G_1, G_2) = \frac{1}{N_m} \sum_{i \in A_{12}} w(c_i) \quad (3)$$

where  $N_m$  is the number of objects as stated before,  $w_i$  the weight of the object  $i$  that is a function of the class  $c_i$ . Averaging over all objects allows to measure the dynamics of the scene, that is what we want to penalize.

*c) Object observability:* Another important feature is the length of the path along which an object is visible. It is preferred that they remain observable on a long range so that the robot can use them as landmarks. Each object is indexed in a table with the number of submaps where it is observable. Taking the maximum of this table as  $o_{max}$  for normalization, we compute the *Observability Score*  $O$  of an edge as follows:

$$O(G_1, G_2) = \frac{1}{N_m} \sum_{i \in A_{12}} (1 - \exp(1 - \frac{o_{max}}{o_i})) \quad (4)$$

where  $o_i$  is the individual observability of each object.

*d) Object Spatial Repartition:* It is important to observe objects in all directions in the spherical image in order to constrain the pose, all ideally by finding the same amount of objects in all directions. We model this ideal repartition by a uniform probability distribution of finding objects with respect to the direction of observation, denoted as  $P$ . Then the image is cut in four parts corresponding to *front*, *right*, *left* and *back*. We count the number of objects in each direction and divide by the total, building the actual repartition function, denoted as  $Q$ . The penalty of using the actual repartition instead of the ideal repartition is computed using the Kullback-Leibler divergence of  $Q$  with respect to  $P$ :

$$KL(Q||P) = \sum_{i=1}^4 (P(i) \cdot \log(\frac{P(i)}{Q(i)})) \quad (5)$$

where the sum is performed on the four parts of the image. Finally the *Repartition Score*  $R$  is computed as:

$$R = 1 - \exp(-KL(Q||P)) \quad (6)$$

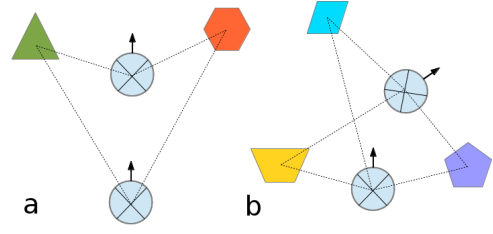


Fig. 2. Illustration of the path description for two consecutive images in the case of translation (a) and composed movement (b). The spherical image is illustrated by a blue circle with a cross highlighting the four quarters. Each object appearing in a different quarter allows to describe the movement of the robot. In example (a), the generated constraints are: move forward until green object  $\in Q_{left}$  and orange  $\in Q_{right}$

Finally the cost of a path  $p$ , made of  $N$  edges  $e_i$  is defined as follows:

$$F_s(p) = \sum_{i=1}^N S(e_i) + C(e_i) + O(e_i) + R(e_i) \quad (7)$$

Summing over all edges allows to implicitly take into account the length of the path. With all the weights in the global graph having been computed, the best path, from the semantic perspective, is found using the well-known Dijkstra's algorithm.

## V. SEMANTIC ROUTE DESCRIPTION

Once the best route is selected within the graph, it can be useful to automatically describe it for human-robot interaction. Many works have addressed the problem of describing the route in terms of high-level behaviour of the robot and semantic places sequence (see section II). Despite slight differences between them, all these works rely on the concept of place that is straightforward indoors, but difficult to define outdoors. In this part, we propose another approach to describe the route at a high level without using places so that our work can easily adapt to any diverse environment. Additionally, our approach naturally alleviates labelling errors by preferring stable objects for which label confidence is higher.

### A. Route Description Language

A route is given as an ordered sequence of local submaps  $\mathcal{S} = \{s_i\}$ . As previously stated, semantic edges connecting submaps encode the object set shared by the two corresponding submaps and they can be described in terms of objects features.

To describe the path at a high level, the transition  $t_{i \rightarrow i+1}$  between  $s_i$  and  $s_{i+1}$  is decomposed in two elements:

- The movement type: turn or move forward/backward
- Its amplitude: do movement until

Each image is cut in four quarters  $Q_i$  corresponding to front, back, right and left. The front is set in the center of the spherical image, corresponding to the image of the first camera on the camera ring of our sensor. A given orientation is described semantically by giving the quarter where each object center belongs to. This results in a set of *Orientation*

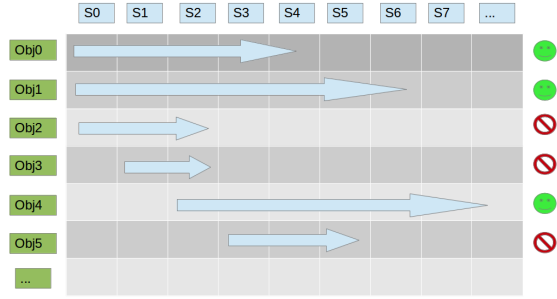


Fig. 3. Index of the objects present for each submap. Observability correspond to the length of arrows. Object 1 and 4 are interesting as they are observed many times while object 3 and 5 are not.

constraints of the form  $\{\text{object } 1 \in Q_{\text{front}}, \text{object } 2 \in Q_{\text{right}}, \text{object } 3 \in Q_{\text{back}}, \dots\}$ .

To estimate the relative movement between two images, the quarters where objects lie are compared in two consecutive images as in fig 2. Rotation shifts objects in the same direction while translation induce different apparent movements.

The amplitude of each action is then set by giving stop conditions, expressed as a set of *Stop constraints* of the form  $\{\text{building} \in Q_{\text{front}}, \text{tree} \in Q_{\text{left}}, \dots\}$ . In fig 2 b for example, robot turns left until blue object  $\in Q_{\text{left}}$ , purple  $\in Q_{\text{right}}$  and yellow  $\in Q_{\text{back}}$ . Stop conditions can use observed objects or any change in the shared objects set. For example a new object appearing in the field of view or an object disappearing can be used to add constraints to the set of *Stop constraints*.

### B. Description Compression

Many objects are visible along the path but they are not all equally important in describing the route. Some objects are visible for a very short time and are not useful to navigate. Additionally, the constraints sets can be similar for successive poses on a long range if the same objects are observed and the movement is constant. Then we may compress the route description using only objects visible on a long range and keeping the same set of constraints. It is worth noting that using only objects with good observability increases the robustness of our approach. In fact, errors in the labelling process produce objects with very small observability and these objects are filtered by this selection process. As each object is tracked along the path, it is easy to know how long a given object will be visible. To choose the best constraint we refer to the index used to compute the observability score (see fig 3). For a given set of constraints, we keep only those referring to objects with the best observability. Instead of computing new constraints at each step, the existing constraints are checked. While they are valid no new constraint is added to the set.

## VI. EXPERIMENTS

We propose two separate series of experiments to evaluate our semantic route planning and our automatic description

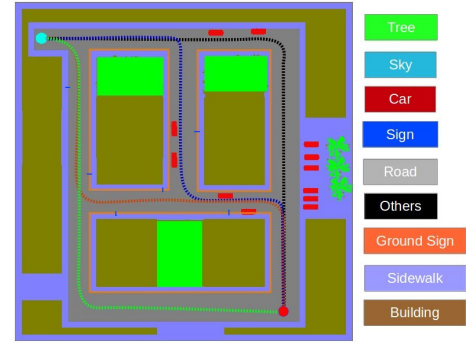


Fig. 4. Map of the simulated world. The four paths are highlighted by the four dashed lines. The black line is the path 1 along which robot meets lots of cars and objects. Following path 2, the blue line, the robot meets lots of objects among which cars and a few sign panels. The third path, materialised in orange, allows the robot to see several sign panels and a few cars. Along the last path, in green, robot sees only a few objects, mainly sign panels. The light blue point is the starting position and the red dot the target one.

framework. For both, we first use a simulated world in which a virtual multi-camera sensor captures  $360^\circ$  views of the environment. We use simulated data as it is more convenient to generate specific cases to highlight the performance and the particular aspects of our algorithm. Then, we show that our algorithm is fully functional in a real-world environment using a challenging dataset where images are automatically labelled with our previous approach presented in [3].

### A. Semantic-based route planning in a virtual world

Our first set of tests is performed on a dataset acquired with a virtual multi camera sensor. Our aim is to evaluate the behaviour of our algorithm depending on environmental features such as the number and class of objects. Our virtual world corresponds to an urban environment with buildings, roads, side-walks, trees, cars, sign panels and ground signs. In our first experiment, the robot is placed at a position from where four paths are possible to reach the target position (see fig 4). Along the first path, the robot encounters many objects, mainly cars, so that it could understand that the scene is highly dynamic. Along the second path, the robot can see a lot of objects with several cars and a few sign panels. The third path is characterized by a high number of sign panels and only a few cars. By this way there is a lot of objects and their mean class value is high. Finally the fourth path offers only a few objects to see, mainly sign panels. To be coherent with an urban scenario, the number of buildings and trees are more or less the same along each path.

Results of the first experiment are shown in the table I. The sum of the four scores for all the edges along each path are reported together with the relative score of each. The preferred path is the third one, as this path offers the best compromise between number of objects and dynamic of the scene. There is a large number of objects and most of them are static so the robot can progress safely. In contrast, the first path contains more objects but a significant part of them are cars and the robot strongly penalizes it. Additionally objects are seen on a short period of time producing a



high *Observability score*. However, as this path contains the bigger number of objects its *Repartition Score* is the best. Although a small number of objects are visible, the last path has a low score as there are no dynamic objects and the linear sections are long enough to provide a small *Observability score*.

TABLE I  
RESULTS FOR SIMULATED WORLD.

Score	$S$	$C$	$O$	$R$	Relative total score
Path 1	19	102	93	<b>148</b>	100
Path 2	21	85	91	153	98
Path 3	<b>18</b>	<b>81</b>	<b>88</b>	149	<b>92</b>
Path 4	19	<b>81</b>	90	158	95

The experiment shows that robot prefers paths with low dynamic, longer straight sections and more objects. However, the number of objects could conflict with the requirement for a minimum number of dynamic objects. In fact the *Repartition Score*(penalty) decreases with the number of objects but the *Class Score* increases. It comes from the fact that areas with large number of objects contain more dynamic objects. Then the choice of the best path is a question of tradeoff between different scores. The tuning of the parameters  $w_i$  allows us to adapt the robot's behaviour to different environments. This result is very close to what it is expected from human behaviour.

### B. Semantic-based route planning in the real-world

Our second set of experiments is performed with a challenging real-world images dataset acquired on the INRIA campus. The environment is semi-urban, with a lot of trees and buildings, cars, side-walk, sign panels and ground signs on the road. Due to the configuration of the campus it is not possible to choose between multiple paths to go from one point to another. Consequently our test consists in evaluating the path weight given different starting positions in order to reach the same target. We also report the length of each path, computed as the sum of distances between each consecutive submaps. The path 1 and 2 are illustrated in fig 5 by blue and red lines respectively. The first one is a quasi-straight line from the starting to the target position. They are two sections with cars parked in front of building on one or two sides of the road. Along the second path the robot moves in a constantly changing environment with several turns, alternating places with trees, buildings and cars parked in several places. Places with cars are highlighted in fig 5 by dotted circles.

Results are presented in table II. The best path according to semantic criteria is the first one, even if it is slightly longer. This could be understood looking at the repartition of objects along the path. In the path 1 there are a lot of cars but they are concentrated in a two places. That way, they appear as a few semantic areas and the robot considers the dynamics of path 1 lower than dynamics of path 2, where cars are spread all along the trajectory and are identified individually. Furthermore, the first path is a straight line and it is possible

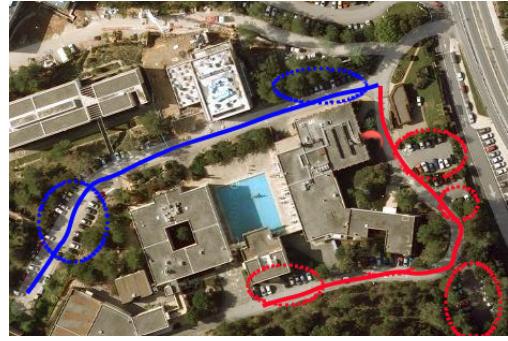


Fig. 5. Example images of the real paths. The blue line (bottom) correspond to path 1 and the red line (top) to path 2. Cars are localized by dotted circles.

to track objects on a long range. It results in a considerably better observability than in path 2. These results validate our approach in real outdoor environments, showing its ability to discriminate between the two routes in a human-friendly way. It shows that path length is not the best choice for navigation as the most dynamic path would have been chosen with classical approaches.

TABLE II  
RESULTS FOR REAL WORLD.

Score	$S$	$C$	$O$	$R$	Relative score	Relative length
Path 1	49	<b>53</b>	<b>58</b>	<b>91</b>	<b>92</b>	100
Path 2	49	65	65	93	100	<b>97</b>

### C. Automatic Route Description

We have evaluated the performances of our automatic route description framework through two parameters: the *correctness* of the produced descriptions and their *compactness*. Obviously, it is not possible to report the full description of a long path, so for the sake of simplicity, the correctness has been evaluated on a small simulated scene. Then the compactness, defined as the number of constraints sets used for a given number of keyframes, has been evaluated on a subset of the INRIA dataset, representing two very different paths, each composed of 200 keyframes.

1) *Correctness of the description*: The path consists of a straight line represented at fig 6. The dataset consists of 100 reference images. Along the path the robot generates three sets of constraints at the positions illustrated by the three numbers. The dotted lines highlight objects used by robot.

The generated constraints are:

- **Orientation**: building (1) left, sign (1) front,  
**Move forward until**: car (1) left,
- **Orientation**: building (1) left, sign (2) front, car (1) left  
**Move forward until**: building (1) not visible, sign (2) right
- **Orientation**: building (2) back, sign (1) back, building (3) front, sign

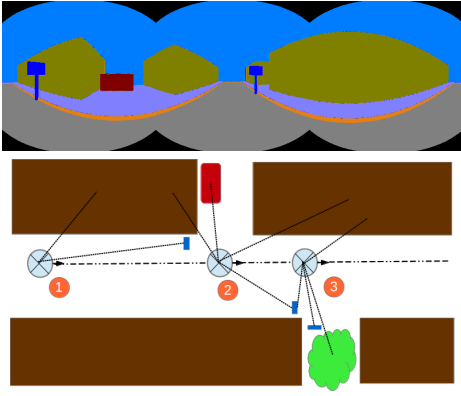


Fig. 6. Top: spherical view of the scene taken from position 2. Bottom: schematic view of the environment. Path 1 in simulated data. Robot is materialized by a crossed circle with an arrow, cars are red, trees green, building brown, panels blue. The robot uses 3 sets of constraints to describe the path at the position highlighted the figures.

(2) right, sign (3) right

**Move forward until:** end

As expected, the path is described with a few constraints and is understandable by human. Despite the simplicity of the example, the result shows the ability to produce human-like description of a path, which is our goal.

2) *Compactness of the Description*: To evaluate the compactness of the description, two different paths have been chosen in INRIA dataset. For each, constraints sets have been generated according to environment features. The results are best viewed in the video<sup>1</sup>. For the first route, a quasi-straight line visible at fig 7 left, the robot has generated 7 constraints sets for 200 reference images. For the second path, visible at fig 7 right, the robot has generated 17 constraints sets for 200 reference images. The compactness are respectively 7/200 and 17/200. The large difference between those two results can be explained by the presence of many occlusions in the second case, that prevent from tracking objects on long range and then actualise constraints sets regularly. However, even if the compactness is lower than in the previous case, the number of constraints is still low. These results show that despite imperfect images annotations, it is possible to produce compact human-like descriptions of trajectories selected according to high-level criteria, in real-world scenes.

## VII. CONCLUSION AND FUTURE WORK

A new approach to automatically select the best route has been presented. The use of semantic content of the scene to choose among several possible paths has been shown to be useful in complex environments where many factors influence robot behaviour. Both simulated and real-world dataset have been used to show the utility of the system. Furthermore a convenient approach for high-level route description has been introduced based on semantic graphs. Both contributions extend the state-of-the-art methods to outdoor environments and open new perspectives towards intelligent navigation and human-robot interaction.

<sup>1</sup><http://youtu.be/JtuGG-LSIJg>



Fig. 7. The two paths used to test automatic route description. At the top is the path 1, below is the path 2.

Extensions of our work can be sought in several directions. Objects are tracked locally in successive images and occlusions can temporally hide an object. Tracking objects not only in successive images but in several neighbours can overpass this problem. Path description and planning would benefit from these improvements. Finally, taking into account the long-term evolution of the environment by monitoring changes can improve path planning process providing the robot with more information about dynamic aspects of the scene.

## ACKNOWLEDGMENT

The work presented in this paper was supported by the French Foundation of technological Research under the grant CIFRE N2012/0067.

## REFERENCES

- [1] A. Borkowski, B. Siemiatkowska, and J. Szklarski, "Towards semantic navigation in mobile robotics," in *Graph Transformations and Model-Driven Engineering*, ser. Lecture Notes in Computer Science, G. Engels, C. Lewerentz, W. Schafer, A. Schrr, and B. Westfechtel, Eds. Springer Berlin Heidelberg, 2010, vol. 5765, pp. 719–748.
- [2] L. F. Posada, F. Hoffmann, and T. Bertram, "Visual semantic robot navigation in indoor environments," in *ISR/Robotik 2014*, June 2014.
- [3] R. Drouilly, P. Rives, and B. Morisset, "Fast hybrid relocation in large scale metric-topologic-semantic map," in *IROS 2014*, 2014.
- [4] C. Galindo, J.-A. Fernandez-Madrigal, J. González, and A. Saffiotti, "Robot task planning using semantic maps," *RAS*, vol. 56, no. 11, pp. 955 – 966, 2008, semantic Knowledge in Robotics.
- [5] U. Klaus, R. Arne, and D. Rudiger, "From structure to actions: Semantic navigation planning in office environments," in *IROS 2011 Workshop on Perception and Navigation for Autonomous Vehicles in Human Environment*.
- [6] C. Landsiedel, R. De Nijs, K. Kuhnlenz, D. Wollherr, and M. Buss, "Route description interpretation on automatically labeled robot maps," in *ICRA 2013*, May 2013, pp. 2251–2256.
- [7] C. Matuszek, D. Fox, and K. Koscher, "Following directions using statistical machine translation," in *International conference on Human-robot interaction*. IEEE Press, 2010, pp. 251–258.
- [8] M. Meilland, A. I. Comport, and P. Rives, "A spherical robot-centered representation for urban navigation," in *IROS 2010*. IEEE, 2010.
- [9] M. Meilland, A. Comport, and P. Rives, "Dense visual mapping of large scale environments for real-time localisation," in *IROS 2011*, sept. 2011.
- [10] W. E. L. Grimson, *Object Recognition by Computer: The Role of Geometric Constraints*. Cambridge, MA, USA: MIT Press, 1990.